

## **For assignment3, we worked in the group of 2.**

**Purui yi:** (complete 50% for the assignment3)

Web design for login and register pages (includes login.css, login.html, register.css, register.html, login.js, register.js, base.html).

complete the postmark.html, feedback.html, success.html, and `@app.route('/postmark')` in the "app.py".

Modify the CSS of the site.

**Feifei fu:** (complete 50% for the assignment3)

Link the flask and database.

Complete `@app.route('/register')`, `@app.route('/login')`, `@app.route('/marks')`, `@app.route('/remark')`, `@app.route('/feedback')`, `@app.route('/feedbacks')`, `@app.route('/logout')` in the "app.py".

Complete the marks.html, feedback.html.

## **We have complete the function of:**

1. User login with username/password with different users (student, instructor).
2. Register a new user. You can register as two types (student, instructor).
3. When login in as a student, you can see your marks by clicking a link, and this mark is the mark stored in the database.
4. Students can submit a remarking request.
5. Students submit an anonymous feedback to the specific instructor.
6. Instructors can see all the grades of his students by clicking on a special link.
7. Instructor can see all the anonymous feedback that was assigned to him/her by clicking on a special link.
8. The instructor can view all regrade requests, and if you manually change them in the database, the front end will reflect that change.
9. Instructors can enter marks for students, and these marks are entered correctly and recorded in the database.

\* Most of our tests were successful, but occasionally we could not find a table in the database when running app.py, but when we changed the data to an absolute path, everything worked fine. So if this happens, please only deduct marks for this part, the rest of our page is intact.

Herokuapp: <https://cscb20a3-demo.herokuapp.com/>

